

COMP 543: Tools and Models for Data Science [Graduate Level]

Description

This course is an introduction to modern data science. Data science is the study of how to extract actionable, non-trivial knowledge from data. The course will focus on the software tools used by practitioners of modern data science, the mathematical and statistical models that are employed in conjunction with such software tools and the applications of these tools and systems to different problems and domains. On the tools side, we will cover the basics of relational database systems, as well as modern systems for manipulating large data sets such as Hadoop MapReduce, Apache Spark, and Google's TensorFlow. On the models side, the course will cover standard supervised and unsupervised models for data analysis and pattern discovery.

Course Objectives

At the end of this course, students will understand the development and use of modern machine learning tools, such as Spark and TensorFlow and will be able to implement machine learning algorithms using these tools. They will have basic skills in querying relational databases and will understand and be able to implement and use common data science models, including gradient descent, logistic regression, outlier detection, deep neural nets and more.

Prerequisites: Mathematical sophistication (calculus, statistics) and programming skills that would be acquired in an undergraduate computer science program are expected. Most programming will be in Python and SQL (SQL is covered in the course), with some Java.

Instructors: Chris Jermaine (cmj4@rice.edu) & Risa Myers (rbm2@rice.edu)

Textbook

There is no textbook for the class. All class material will be conveyed during lecture.

Meeting Times and Locations

Class will be held Monday, Wednesday, & Friday from 1 PM to 1:50 PM in Duncan Hall 1075.

Registration

For Spring 2018, permission of the instructor is required for this course. Please email Risa with your request.

Syllabus

The tentative lecture schedule is:

1. Comp 543 Introduction
2. Intro to relational databases 1
3. Intro to relational databases 2
4. The relational algebra
5. Declarative SQL 1
6. Declarative SQL 2
7. Declarative SQL 3
8. Imperative SQL 1
9. Imperative SQL 2
10. Hadoop and MapReduce
11. Intro to modeling: numerical s. probabilistic vs. Bayesian 1
12. Intro to modeling: numerical s. probabilistic vs. Bayesian 2
13. Optimization basics: Gradient descent
14. Optimization basics: Newton's method
15. Optimization basics: Expectation maximization
16. Python and Data Science
17. Intro to supervised learning
18. Intro to unsupervised learning
19. Linear regression
20. Data analytics programming on spark: python in the cloud
21. Generalized linear models
22. Support Vector Machines
23. Over-fitting and regularization
24. Sequential models
25. Mixture models
26. Outliers
27. Dimensionality reduction
28. Deep Neural Networks
29. Deep Neural Networks 2: Learning
30. Deep Neural Networks 3: RNN
31. Deep Neural Networks 4: LSTM
32. Deep Learning with TensorFlow

Communication

The class will have a Piazza forum for all day-to-day communication:

<https://piazza.com/rice/spring2018/543/home>

It is expected that if you have a technical question on an assignment, you will post it to the forum rather than sending an email to the instructors. This approach guarantees a fast response and means that everyone can benefit from the question and the answer. In general, only inquiries of private or personal nature should be made directly to the instructors ("I need to go out of town on Oct 22nd, can I have an extra day..."). Everything else should be posted on Piazza. You'll get faster feedback from the group than you can get from your instructors.

If you have any communication of a more personal nature and wish to contact the instructors of the class, please send an email to Chris & Risa, and include the word “543” in the subject line. Please realize that we get a lot of random email, so if you do not include 543 in the subject line, your email will likely be ignored.

Assignment handouts and turnins, as well as your grades, will be on Canvas. Everything else will be on Piazza.

Class Attendance

Attendance will not be taken in class, but you are highly encouraged to attend and participate in class.

Grading and Evaluation

Your grade is based upon a set of programming assignments (60% of your grade; each is worth 10% of your grade), lab meetings (10% of your grade), Research summaries (10% of your grade: the first two are 1% each, the next four are worth 2% each) and four written homework assignments (5% of your grade each).

Final grades are based on the numeric grades, where 90-100 is an A, 80-89 is a B, and so forth. We reserve the right to apply a “curve” to change this, but only for the better. That is, if you’ve gotten 90%, you’re guaranteed at minimum an A- for your final grade, but you might do better.

Assignments

This is an assignment-oriented class. There will be six programming assignments, all completed individually. All programming assignments will be in SQL (A1 and A2), Java (A3) and Python (A4, A5, A6).

The assignments are expected to be:

Assignment #	Description	Out	In
A1	SQL programming	01/24/2018	02/03/2018
A2	Data Analytics in SQL	02/04/2018	02/17/2018
A3	Hadoop Programming	02/18/2018	03/07/2018
A4	Intro to Spark Programming	03/19/2018	04/02/2018
A5	Linear Regression on Spark	04/04/2018	04/16/2018
A6	Deep Learning on TensorFlow	04/16/2018	Date of class final

Homeworks

The homeworks are expected to be:

Homework #	Description	Out	In
HW1	Relational Calculus and Relational Algebra	01/17/2018	01/26/2018
HW2	Numerical Optimization / gradient descent	02/07/2018	02/14/2018
HW3	Expectation Maximization	02/17/2018	02/24/2018
HW4	Outliers	03/21/2018	03/28/2018

Readings/Research Summaries

Students are required to read seminal papers on each of the systems and algorithms as well as 1-2 additional research papers on the use of the system or algorithm as applied to a domain of personal interest. Students will submit a 1 page related work summary for each area.

The readings are expected to be:

Reading #	Asmt #	Papers	Due Date
R1	A1/HW1	1. EF Codd. A Relational Model of Data for Large Shared Data Banks. Commun ACM, 13(6) June 1970, pp. 377-387	02/07/2018
R2	A2	1. L Page, S. Brin, R Motwanim T Winograd. "The PageRank Citation Ranking: Brining Order to the Web" Stanford Digital Library Technologies, January 1998	02/21/2018
R3	HW3	1. N Friedman, "The Bayesian Structural EM Algorithm" UAI 1998.	02/28/2018
R4	A3	1. S Ghemawat, H Gobioff, ST Leung. "The Google File System" SOSP 2003 37(5) October 2003, pp. 29-43. 2. J Dean, S Ghemawat. "MapReduce: Simplified Data Processing on Large Clusters" Commun ACM 51(1) Jan	03/23/2018

		2008, pp. 107-113.	
R5	A4	1. M Zaharia, M Chowdhury, MJ Franklin, S Shenker, I Stoica. "Spark: Cluster Computing with Working Sets" Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, 2010. 2. TM Cover, PE Hart. "Nearest Neighbor Pattern Classification" IEEE Transactions on Information Theory 13(1) 1967, pp. 21-27.	04/09/2018
R6	A6	1. M Abadi, A Agarwal, P Barham, E Brevdo et al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems" arXiv:1603.04467, 2016.	04/20/18

Late Assignment Policy

Assignments must be turned in by 11:55PM (5 minutes before midnight) on the day that they are due. You can turn in an assignment up to 24 hours late, in which case you receive a 10% penalty (that is, 10 points are subtracted from an assignment that is worth 100 points), or up to 48 hours late, in which case you receive a 20% penalty. Assignments turned in after that time are not accepted. Please note that your turnin time is whatever Canvas says, and your turnin is whatever you turn into Canvas, **no exceptions**. Be safe; submit early and often!

Though we say **no exceptions**, there are exceptions in very extreme circumstances, with proper documentation. For example, if you obtain a doctor/dentist note stating that you were so ill at the due date/time that you could not reasonably be expected to meet the deadline, it is possible to get an extension.

Due date extensions may be requested up to one week prior to the assignment/homework/reading due date. After that, extensions will be granted only under extreme circumstances, with proper documentation, as described above.

Regrade Requests

These must be made within **one week** of an assignment/homework being returned, during Chris' or Risa's office hours, to Chris or Risa in person. Sending an email does not constitute a regrade request. When you talk to Chris or Risa, the instructor will help you understand whether you've got a legitimate request. If you do, then you'll write that request down formally, print it on paper, and hand it to Chris or Risa. Chris & Risa will batch these, and then periodically and issue final grade adjustments in bulk for everybody.

Rice Honor Code

In this course, all students will be held to the standards of the Rice Honor Code, a code that you pledged to honor when you matriculated at this institution. If you are unfamiliar with the details of this code and how it is administered, you should consult the Honor System Handbook at <http://honor.rice.edu/honor-system-handbook/>. This handbook outlines the University's expectations for the integrity of your academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty members throughout the process.

Academic Misconduct

In a programming class, there is sometimes a very fine line between "cheating" and acceptable and beneficial interaction between peers. Thus, it is very important that you fully understand what is and what is not allowed in terms of collaboration with your classmates. Our goal here is to be 100% precise, so that there can be no confusion.

The rule on collaboration and communication with your classmates is very simple: you cannot transmit or receive code from or to anyone in the class in any way---visually (by showing someone your code), electronically (by emailing, posting, or otherwise sending someone your code), verbally (by reading code to someone) or in any other way we have not yet imagined. Any other collaboration is acceptable.

The rule on collaboration and communication with people who are not your classmates (or your TAs or instructor) is also very simple: it is not allowed in any way, period. This disallows (for example) posting any questions of any nature to programming forums such as StackOverflow.

As far as going to the web and using Google, we will apply the "two line rule." Go to any web page you like and do any search that you like. But you cannot take more than two lines of code from an external resource and actually include it in your assignment in any form. Note that changing variable names or otherwise transforming or obfuscating code you found on the web does not render the "two line rule" inapplicable. It is still a violation to obtain more than two lines of code from an external resource and turn it in, whatever you do to those two lines after you first obtain them. Furthermore, you should **cite your sources**. Add a comment to your code that includes the URL(s) that you consulted when constructing your solution. This turns out to be very helpful when you're looking at something you wrote a while ago and you need to remind yourself what you were thinking.

Any violations of these rules will be reported to the Honor Council. Just don't do it!

Students with Disabilities

If you have a documented disability that may affect academic performance, you should:

- 1) make sure this documentation is on file with Disability Support Services (Allen Center, Room 111 / adarice@rice.edu / x5841) to determine the accommodations you need; and
- 2) meet with the instructors to discuss your accommodation needs.